In [1]:

**import pandas as pd import numpy as np import seaborn as sns**

**import matplotlib.pyplot as plt**

In [2]:

nesarc = pd.read\_csv('nesarc.csv', low\_memory=**False**) pd.set\_option('display.float\_format', **lambda** x:'**%f**'%**x**)

# From Prac 1

**Columns/Data used in Prac 1**

In [3]:

nesarc['S2AQ5B'] = pd.to\_numeric(nesarc['S2AQ5B'], errors='coerce') *#convert variable t o numeric*

nesarc['S2AQ5D'] = pd.to\_numeric(nesarc['S2AQ5D'], errors='coerce') *#convert variable t o numeric*

nesarc['S2AQ5A'] = pd.to\_numeric(nesarc['S2AQ5A'], errors='coerce') *#convert variable t o numeric*

nesarc['S2BQ1B1'] = pd.to\_numeric(nesarc['S2BQ1B1'], errors='coerce') *#convert variable to numeric*

nesarc['AGE'] = pd.to\_numeric(nesarc['AGE'], errors='coerce') *#convert variable to nume ric*

# From Prac 2

**A subset of nesarc data, with the following criteria Age from 26 to 50**

# Beer drinking status - S2AQ5A = Y

In [4]:

sub1=nesarc[(nesarc['AGE']>=26) & (nesarc['AGE']<=50) & (nesarc['S2AQ5A']==1)] sub2=sub1.copy()

# From Prac 2

**SETTING MISSING DATA**

sub2['S2AQ5D']=sub2['S2AQ5D'].replace(99, np.nan)

sub2['S2AQ5B']=sub2['S2AQ5B'].replace(8, np.nan) sub2['S2AQ5B']=sub2['S2AQ5B'].replace(9, np.nan) sub2['S2AQ5B']=sub2['S2AQ5B'].replace(10, np.nan) sub2['S2AQ5B']=sub2['S2AQ5B'].replace(99, np.nan)

sub2['S2BQ1B1']=sub2['S2BQ1B1'].replace(9, np.nan)

# From Prac 2 Recode data

In [6]:

recode2 = {1:30, 2:26, 3:14, 4:8, 5:4, 6:2.5, 7:1}

sub2['BEER\_FEQMO']= sub2['S2AQ5B'].map(recode2)

recode3 = {2:0, 1:1}

sub2['S2BQ1B1']= sub2['S2BQ1B1'].map(recode3)

# From Prac 2

**Create secondary variables**

In [7]:

*# A secondary variable multiplying the number of days beer consumed/month and the appro x number of*

*# beer consumed/day* sub2['NUMBEERMO\_EST']=sub2['BEER\_FEQMO'] \* sub2['S2AQ5D']

# Draw a Line chart

**Age vs Number of beer consumed per month (NUMBEERMO\_EST)**

# mean number of beer consumed

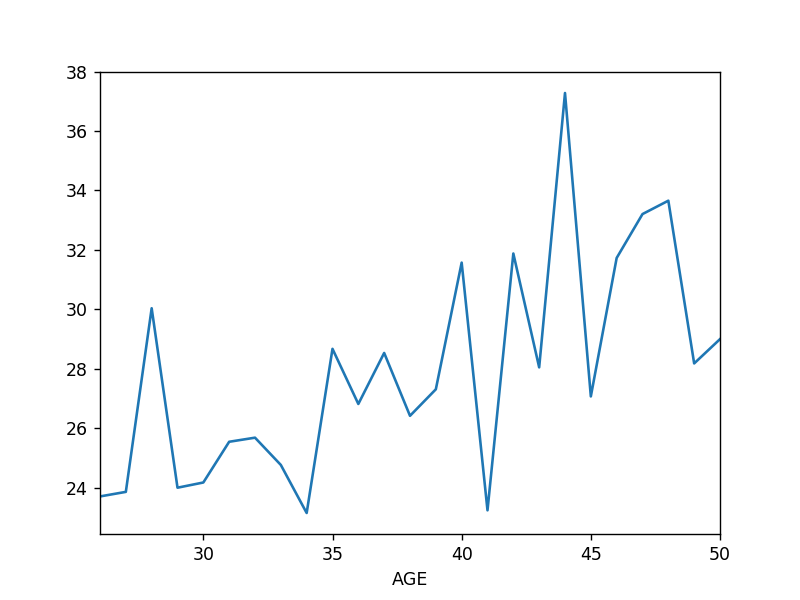
**var = mean number of beers consumed a month, grouped by age**

var = sub2.groupby(['AGE']).NUMBEERMO\_EST.mean() print(var)

*# The output provides the average amount of beer consumed by interviewees within different age groups ranging from 26 to 50. It indicates that most interviewees in this age range drink around 20 to 30 beers per month on average.*

|  |  |
| --- | --- |
| AGE |  |
| 26 | 23.701357 |
| 27 | 23.854545 |
| 28 | 30.035270 |
| 29 | 23.994949 |
| 30 | 24.170530 |
| 31 | 25.541033 |
| 32 | 25.678994 |
| 33 | 24.761017 |
| 34 | 23.143713 |
| 35 | 28.668478 |
| 36 | 26.813272 |
| 37 | 28.530387 |
| 38 | 26.414773 |
| 39 | 27.307122 |
| 40 | 31.571023 |
| 41 | 23.233788 |
| 42 | 31.877676 |
| 43 | 28.045455 |
| 44 | 37.279762 |
| 45 | 27.067241 |
| 46 | 31.727799 |
| 47 | 33.204918 |
| 48 | 33.655303 |
| 49 | 28.177778 |
| 50 | 28.995614 |

Name: NUMBEERMO\_EST, dtype: float64

In [30]:

%**matplotlib** notebook var.plot(kind='line')

*# The line graph illustrates the variation in the number of beer bottles consumed by interviewees between the ages of 26 and 50 on a monthly basis. The graph depicts fluctuations in beer consumption over the specified age range.*

Out[30]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x6c1b8552b0>

# total number of beer consumed

**var2 = sum number of beers consumed a month,**

In [10]:

var2 = sub2.groupby(['AGE']).NUMBEERMO\_EST.sum() print(var2)

*# The result shows the total amount of beer consumed by interviewees every month, categorized by age from 26 to 50. Among this age group, individuals aged 40, 42, and 37 drink the highest amount of beer per month with 11,113, 10,424, and 10,328 bottles, respectively, while the 26-year-old group has the lowest amount with 5,238 bottles consumed every month.*

|  |  |  |
| --- | --- | --- |
| AGE |  |  |
| 26 | 5238.000000 |
| 27 | 6560.000000 |  |
| 28 | 7238.500000 |  |
| 29 | 7126.500000 |  |
| 30 | 7299.500000 |  |
| 31 | 8403.000000 |  |
| 32 | 8679.500000 |  |
| 33 | 7304.500000 |  |
| 34 | 7730.000000 |  |
| 35 | 7912.500000 |  |
| 36 | 8687.500000 |  |
| 37 | 10328.000000 |  |
| 38 | 9298.000000 |  |
| 39 | 9202.500000 |  |
| 40 | 11113.000000 |  |
| 41 | 6807.500000 |  |
| 42 | 10424.000000 |  |
| 43 | 8021.000000 |  |
| 44 | 9394.500000 |  |
| 45 | 7849.500000 |  |
| 46 | 8217.500000 |  |
| 47 | 8102.000000 |  |
| 48 | 8885.000000 |  |
| 49 | 6340.000000 |  |
| 50 | 6611.000000 |  |
| Name: | NUMBEERMO\_EST, | dtype: float64 |

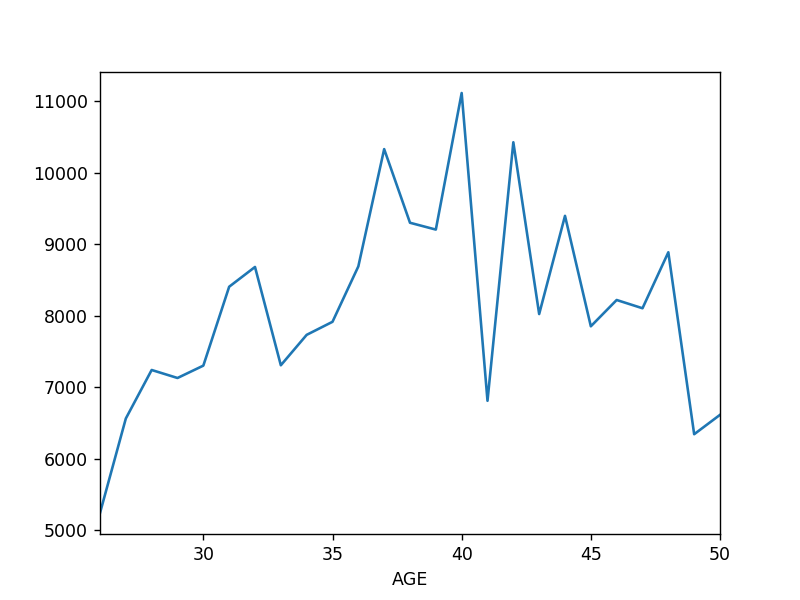
In [31]:

fig = plt.figure() var2.plot(kind='line')

*# The line graph illustrates the monthly beer consumption for each age group from 26 to 50 years old. It provides an overview of the variation in beer consumption among different age groups over time.*

Out[31]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x6c06ab77f0>

# Draw a stacked Column Chart x = age (AGE)

**y = number of beers consumed per month (NUMBEERMO\_EST)**

# stack is based on depedency on beer (S2BQ1B1)

**var3 = mean number of beers consumed a month, grouped by age and beer depedency (S2BQ1B1)**

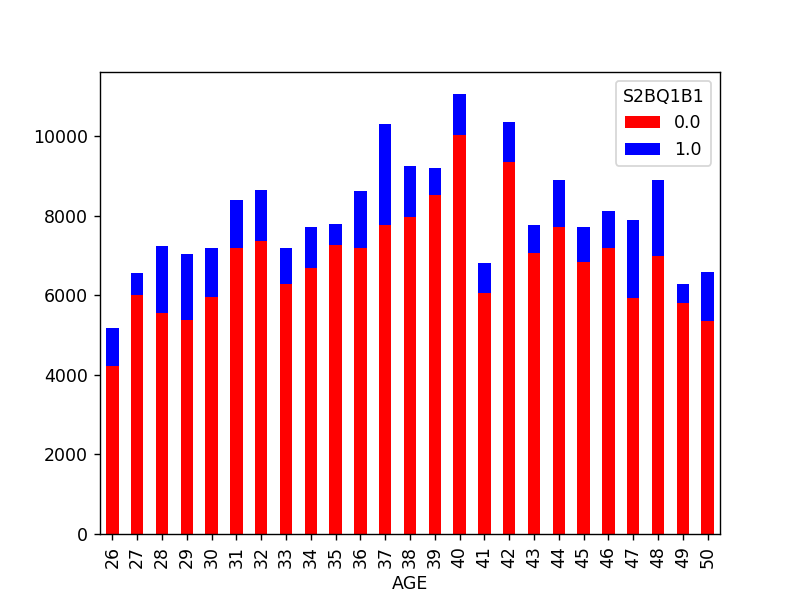
var3 = sub2.groupby(['AGE','S2BQ1B1']).NUMBEERMO\_EST.sum() print(var3)

*# The result provides the total amount of beer consumed per month for each age group (from 26 to 50) and further divides the data based on whether interviewees are alcohol-dependent or not. It shows the variation in beer consumption between alcohol-dependent and non-alcohol-dependent individuals within each age group.*

|  |  |  |
| --- | --- | --- |
| AGE | S2BQ1B1 |  |
| 26 | 0.000000 | 4225.500000 |
|  | 1.000000 | 949.000000 |
| 27 | 0.000000 | 6000.000000 |
|  | 1.000000 | 560.000000 |
| 28 | 0.000000 | 5542.500000 |
|  | 1.000000 | 1686.500000 |
| 29 | 0.000000 | 5363.500000 |
|  | 1.000000 | 1675.000000 |
| 30 | 0.000000 | 5942.500000 |
|  | 1.000000 | 1244.000000 |
| 31 | 0.000000 | 7185.500000 |
|  | 1.000000 | 1212.500000 |
| 32 | 0.000000 | 7352.500000 |
|  | 1.000000 | 1288.500000 |
| 33 | 0.000000 | 6279.000000 |
|  | 1.000000 | 901.500000 |
| 34 | 0.000000 | 6672.000000 |
|  | 1.000000 | 1039.500000 |
| 35 | 0.000000 | 7264.500000 |
|  | 1.000000 | 518.000000 |
| 36 | 0.000000 | 7190.000000 |
|  | 1.000000 | 1420.000000 |
| 37 | 0.000000 | 7765.000000 |
|  | 1.000000 | 2531.000000 |
| 38 | 0.000000 | 7962.000000 |
|  | 1.000000 | 1294.000000 |
| 39 | 0.000000 | 8519.000000 |
|  | 1.000000 | 667.500000 |
| 40 | 0.000000 | 10030.500000 |
|  | 1.000000 | 1022.500000 |
| 41 | 0.000000 | 6047.000000 |
|  | 1.000000 | 755.500000 |
| 42 | 0.000000 | 9352.500000 |
|  | 1.000000 | 986.500000 |
| 43 | 0.000000 | 7061.000000 |
|  | 1.000000 | 695.000000 |
| 44 | 0.000000 | 7711.500000 |
|  | 1.000000 | 1186.000000 |
| 45 | 0.000000 | 6839.000000 |
|  | 1.000000 | 865.500000 |
| 46 | 0.000000 | 7180.000000 |
|  | 1.000000 | 925.500000 |
| 47 | 0.000000 | 5938.500000 |
|  | 1.000000 | 1949.000000 |
| 48 | 0.000000 | 6971.500000 |
|  | 1.000000 | 1913.500000 |
| 49 | 0.000000 | 5799.500000 |
|  | 1.000000 | 491.000000 |
| 50 | 0.000000 | 5341.500000 |
|  | 1.000000 | 1230.500000 |

Name: NUMBEERMO\_EST, dtype: float64

var3.unstack().plot(kind='bar', stacked=**True**, color=['red','blue'], grid=**False**)



*# The bar plot provides a visual representation of the total amount of beer consumed per month by interviewees aged 26 to 50, divided into two groups based on whether they are alcohol-dependent or not. It illustrates the variation in beer consumption between alcohol-dependent and non-alcohol-dependent individuals within each age group.*

Out[32]:

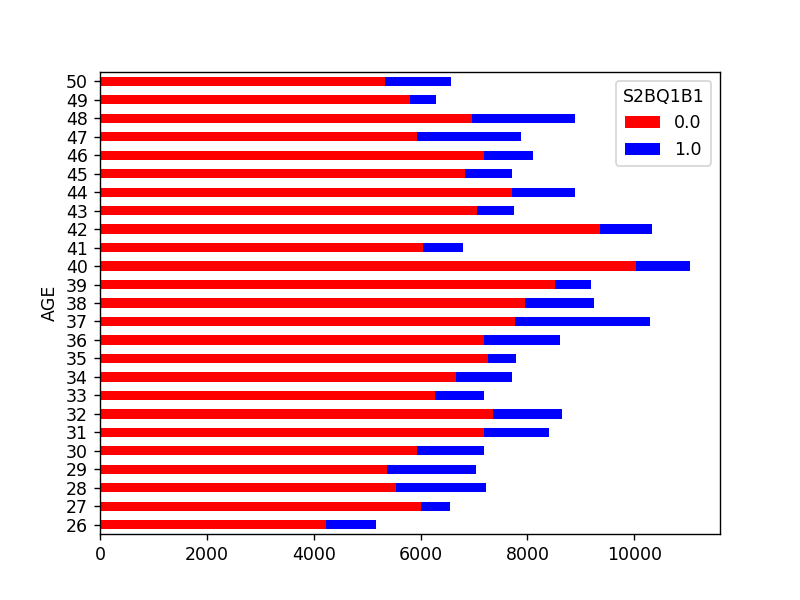
<matplotlib.axes.\_subplots.AxesSubplot at 0x6c0885bd68>

# Draw a horizontal stacked Column Chart x = age (AGE)

**y = number of beers consumed per month (NUMBEERMO\_EST)**

# stack is based on depedency on beer (S2BQ1B1)

var3.unstack().plot(kind='barh', stacked=**True**, color=['red','blue'], grid=**False**)



*# The horizontal bar plot shows the sum amount of beer consumed every month by interviewees from age 26 to age 50, divided into two groups based on whether they are alcohol-dependent or not. It provides a clear comparison of beer consumption between alcohol-dependent and non-alcohol-dependent individuals within each age group.*

Out[33]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x6c14836588>

# Draw a Pie Chart showing age (AGE) and total beer consumed a month (NUMBEERMO\_EST)

**hint use var2**

print(var2)

|  |  |  |
| --- | --- | --- |
| AGE |  |  |
| 26 | 5238.000000 |
| 27 | 6560.000000 |  |
| 28 | 7238.500000 |  |
| 29 | 7126.500000 |  |
| 30 | 7299.500000 |  |
| 31 | 8403.000000 |  |
| 32 | 8679.500000 |  |
| 33 | 7304.500000 |  |
| 34 | 7730.000000 |  |
| 35 | 7912.500000 |  |
| 36 | 8687.500000 |  |
| 37 | 10328.000000 |  |
| 38 | 9298.000000 |  |
| 39 | 9202.500000 |  |
| 40 | 11113.000000 |  |
| 41 | 6807.500000 |  |
| 42 | 10424.000000 |  |
| 43 | 8021.000000 |  |
| 44 | 9394.500000 |  |
| 45 | 7849.500000 |  |
| 46 | 8217.500000 |  |
| 47 | 8102.000000 |  |
| 48 | 8885.000000 |  |
| 49 | 6340.000000 |  |
| 50 | 6611.000000 |  |
| Name: | NUMBEERMO\_EST, | dtype: float64 |

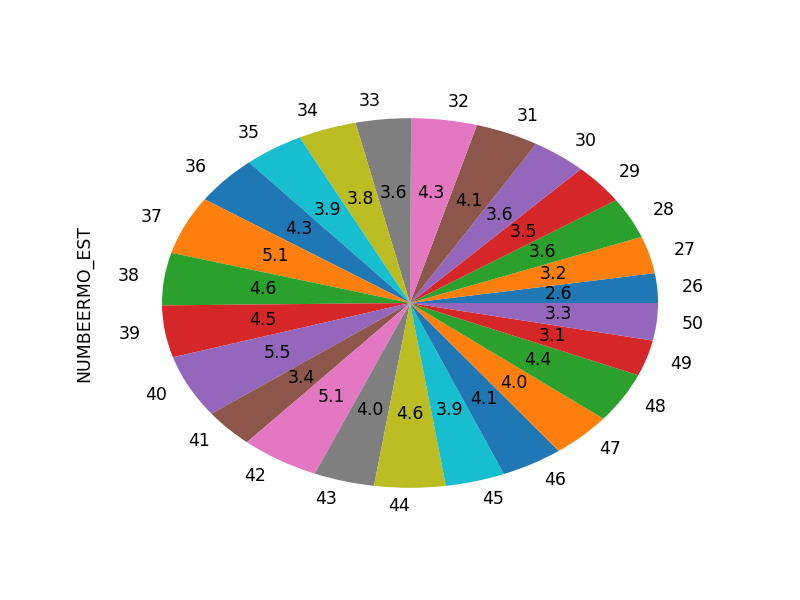
In [34]:

fig = plt.figure() var2.plot(kind='pie',autopct='**%.1f**')

# The pie chart illustrates the distribution of beer consumption among different age groups (from 26 to 50). Each slice represents the proportion of beer consumed by each age group in relation to the total amount of beer consumed across all age groups.

Out[34]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x6c14d07e10>

# Draw a Violin Plot for age (AGE) and income (S1Q10A)

**convert income (S1Q10A) to numeric**

In [17]:

sub2['S1Q10A'] = pd.to\_numeric(nesarc['S1Q10A']) *#convert variable to numeric*

# Plot violin plot

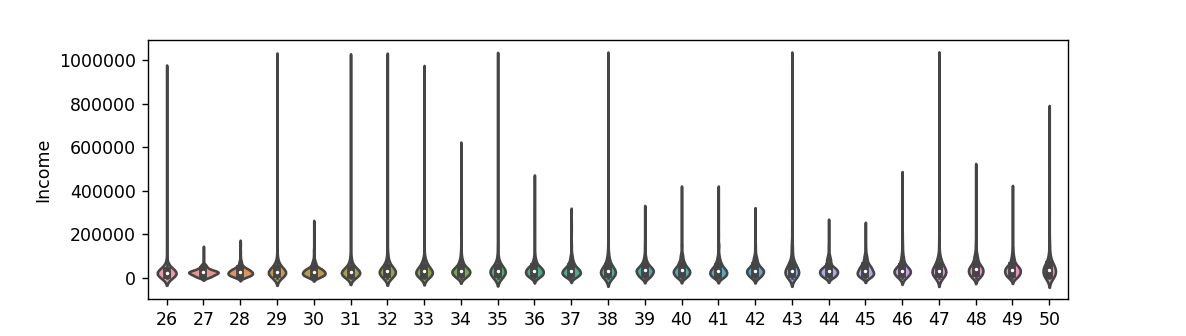
In [35]:

fig = plt.figure()

sns.violinplot(x='AGE', y='S1Q10A', data=sub2) plt.xlabel('Age')

plt.ylabel('Income')

Out[35]:

*# The violin plot displays the distribution of incomes from age 26 to 50. It indicates that there is a slight upward trend in income as the interviewees age, with the distribution widening at older ages, suggesting greater income variability in those age groups.*

Text(0,0.5,'Income')

# Draw a HeatMap for Ethnicity and Carton of Beer consumed per month, based on depedency on beer

**Rename Race - From Module 4**

In [19]:

*# you can rename categorical variable values for graphing if original values are not in formative*

*# first change the variable format to categorical if you haven’t already done so* sub2['ETHRACE2A'] = sub2['ETHRACE2A'].astype('category')

sub2['ETHRACE2A']=sub2['ETHRACE2A'].cat.rename\_categories(["White", "Black", "NatAm", "Asian", "Hispanic"])

# Create a new variable CARTON\_ADAY using CARTON\_ADAY function provided

In [20]:

**def** CARTON\_ADAY (row):

**if** row['BEER\_FEQMO'] >= 30 :

**return** 1

**elif** row['BEER\_FEQMO'] < 30 : **return** 0

sub2['CARTON\_ADAY'] = sub2.apply (**lambda** row: CARTON\_ADAY (row),axis=1)

# Print the size of CARTON\_ADAY, grouped by category

In [21]:

c4= sub2.groupby('CARTON\_ADAY').size() print(c4)

*# The results indicate that 6897 interviewees do not drink more than a carton of beer per day, while 417 interviewees consume more than a carton of beer daily. This information helps distinguish between the two groups based on their beer consumption patterns.*

|  |  |
| --- | --- |
| CARTON\_ADAY |  |
| 0.000000 | 6897 |
| 1.000000 | 417 |
| dtype: int64 |  |

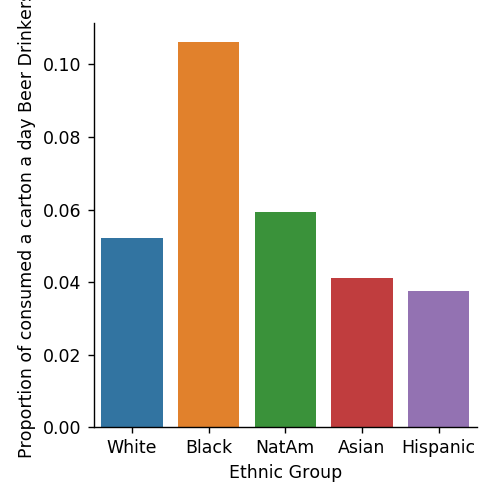
# Draw bar chart to show relationship between race (ETHRACE2A) and CARTON\_ADAY

*# bivariate bar graph C->C*

%**matplotlib** notebook

sns.factorplot(x='ETHRACE2A', y='CARTON\_ADAY', data=sub2, kind="bar", ci=**None**) plt.xlabel('Ethnic Group')

plt.ylabel('Proportion of consumed a carton a day Beer Drinkers')



*# The bar plot displays the proportion of individuals who consume more than a carton of beer per day in five different ethnic groups: White, Black, Native American, Asian, and Hispanic. The plot shows that Black individuals have the highest proportion (0.105) of beer drinkers consuming more than a carton per day, while Asians and Hispanics have lower proportions, around 0.04.*

Out[36]:

Text(0.694444,0.5,'Proportion of consumed a carton a day Beer Drinkers')

# Make copy of just race (ETHRACE2A) and CARTON\_ADAY

sub3 = sub2[['ETHRACE2A','CARTON\_ADAY']].copy()

sub3.head()

Out[23]:

*# The data contains the ethnic groups and whether individuals drink more than a carton of beer per day. The "NaN" values indicate missing data for some individuals' carton per day drinking habits.*

|  |  |  |
| --- | --- | --- |
|  | **ETHRACE2A** | **CARTON\_ADAY** |
| **1** | Hispanic | nan |
| **8** | White | nan |
| **12** | Asian | 0.000000 |
| **16** | White | nan |
| **24** | Hispanic | nan |

# Create pivot table of race (ETHRACE2A) and CARTON\_ADAY

In [24]:

table = pd.pivot\_table(sub3, index=['ETHRACE2A'], columns=['CARTON\_ADAY'], aggfunc=np.s ize)

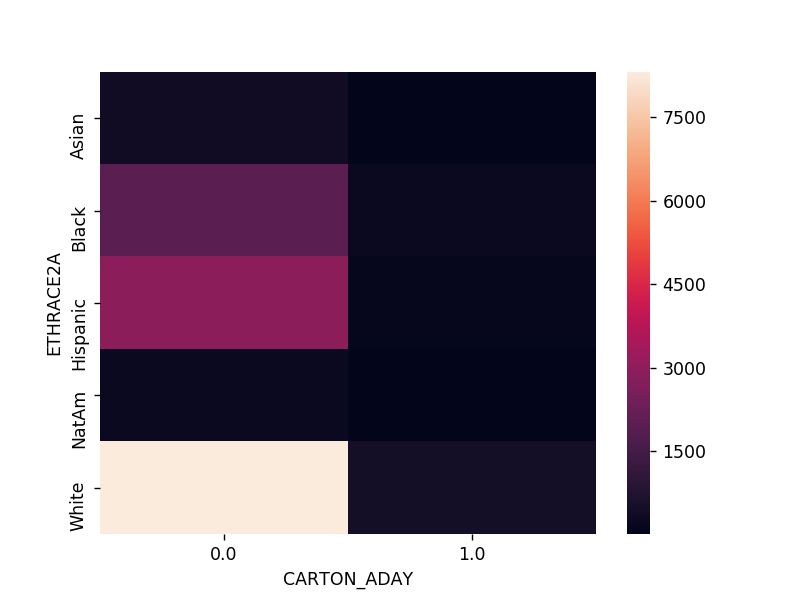
print(table)

*# The pivot table displays the number of individuals from each ethnic group who drink less than a carton of beer per day (0.0) and the number of individuals who drink more than a carton of beer per day (1.0). For example, there are 8312 White individuals who drink less than a carton of beer per day, and 456 White individuals who drink more than a carton of beer per day. Similar information is provided for the other ethnic groups as well.*

|  |  |  |
| --- | --- | --- |
| CARTON\_ADAY  ETHRACE2A | 0.000000 | 1.000000 |
| Asian | 374 | 16 |
| Black | 1972 | 234 |
| Hispanic | 2914 | 114 |
| NatAm | 222 | 14 |
| White | 8312 | 456 |

# Draw heat map

fig = plt.figure() sns.heatmap(table)



*# The heatmap visually represents the number of individuals from each ethnic group who drink less than a carton of beer per day (0.0) and the number of individuals who drink more than a carton of beer per day (1.0). The darker colors indicate higher counts of individuals in each category for each ethnic group.*

Out[37]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x6c05e2a9e8>

# Draw a bubble Chart Read in gapminder.csv

pd.set\_option('display.float\_format', **lambda** x:'**%.2f**'%**x**)

gapminder = pd.read\_csv('gapminder.csv', low\_memory=**False**) gapminder.head()

Out[26]:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **country** | **incomeperperson** | **alcconsumption** | **armedforcesrate** | **breastcancerper** |
| **0** | Afghanistan |  | .03 | .5696534 | 26.8 |
| **1** | Albania | 1914.99655094922 | 7.29 | 1.0247361 | 57.4 |
| **2** | Algeria | 2231.99333515006 | .69 | 2.306817 | 23.5 |
| **3** | Andorra | 21943.3398976022 | 10.17 |  |  |
| **4** | Angola | 1381.00426770244 | 5.57 | 1.4613288 | 23.1 |

*# The table displays the first five rows of the 'gapminder' dataset, showing various statistics for different countries, including income per person, alcohol consumption, breast cancer rate, CO2 emissions, female employment rate, HIV rate, internet usage rate, life expectancy, oil consumption per person, polity score, residential electricity consumption, suicide rate, employment rate, and urbanization rate.*

# Convert internetuserate, urbanrate and incomeperperson to numeric

In [27]:

gapminder['internetuserate'] = pd.to\_numeric(gapminder['internetuserate'],errors='coerc e')

gapminder['urbanrate'] = pd.to\_numeric(gapminder['urbanrate'],errors='coerce') gapminder['incomeperperson'] = pd.to\_numeric(gapminder['incomeperperson'],errors='coerc e')

In [28]:

gapminder\_clean=gapminder.dropna()

# Draw a bubble Chart x = urbanrate

**y = income per person bubble size = internetuserate**

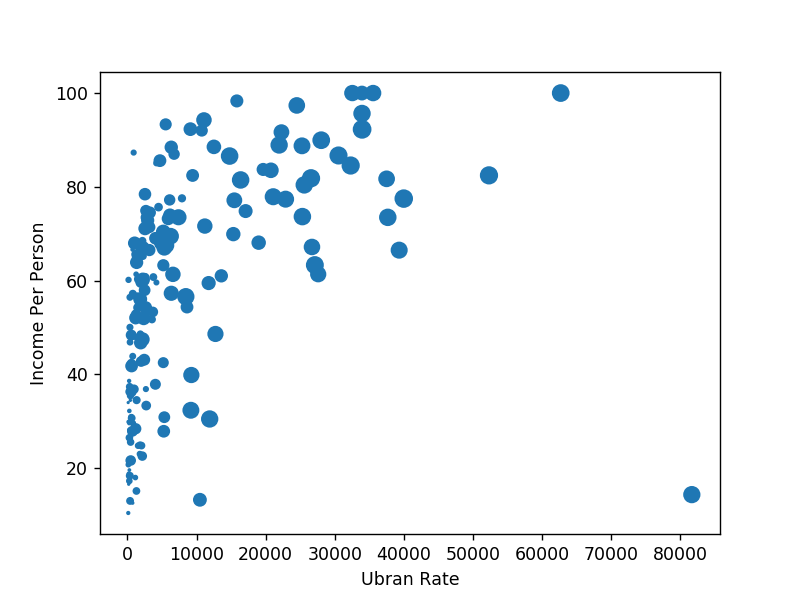
*# x = internetuserate # y = incomeperperson*

*# Added third variable income as size of the bubble*

%**matplotlib** notebook fig = plt.figure()

plt.scatter(gapminder\_clean['incomeperperson'],gapminder\_clean['urbanrate'], s=gapminde r\_clean['internetuserate'])

plt.xlabel('Ubran Rate') plt.ylabel('Income Per Person')



Out[29]:

*# The bubble chart displays the relationship between urbanization rate and income per person, with the size of the bubbles representing the internet usage rate. The chart suggests a positive correlation between urbanization rate and income per person, and larger bubbles indicate higher internet usage rates in those countries.*

Text(0,0.5,'Income Per Person')